


[Subscribe \(Full Service\)](#) [Register \(Limited Service, Free\)](#) [Login](#)

 Search: ☒ The ACM Digital Library ☐ The Guide



THE ACM DIGITAL LIBRARY

[Feedback](#) [Report a problem](#) [Satisfaction survey](#)

 Terms used database and write and buffer and log and audit

Found 40,920 of 184,245

 Sort results  
by

 Display  
results

☒ [Save results to a Binder](#)
☒ [Search Tips](#)
☐ [Open results in a new window](#)

 Try an [Advanced Search](#)

 Try this search in [The ACM Guide](#)

Results 1 - 20 of 200

 Result page: [1](#) [2](#) [3](#) [4](#) [5](#) [6](#) [7](#) [8](#) [9](#) [10](#) [next](#)

Best 200 shown

 Relevance scale ☐ ☐ ☐ ☐ ☐

### 1 [801 storage: architecture and programming](#)



Albert Chang, Mark F. Mergen

 February 1988 **ACM Transactions on Computer Systems (TOCS)**, Volume 6 Issue 1

Publisher: ACM Press

 Full text available: [pdf\(1.87 MB\)](#)

 Additional Information: [full citation](#), [abstract](#), [references](#), [citations](#), [index terms](#), [review](#)

Based on novel architecture, the 801 minicomputer project has developed a low-level storage manager that can significantly simplify storage programming in subsystems and applications. The storage manager embodies three ideas: (1) large virtual storage, to contain all temporary data and permanent files for the active programs; (2) the innovation of database storage, which has implicit properties of access serializability and atomic update, similar to those of ...

### 2 [A comparative study of log-only and in-place update based temporal object database systems](#)



Kjetil Nørvgå

 November 2000 **Proceedings of the ninth international conference on Information and knowledge management**

Publisher: ACM Press

 Full text available: [pdf\(231.70 KB\)](#) Additional Information: [full citation](#), [references](#), [index terms](#)

### 3 [A component model for standardized web-based education](#)


 August 2001 **Journal on Educational Resources in Computing (JERIC)**

Publisher: ACM Press

 Full text available: [pdf\(384.31 KB\)](#) Additional Information: [full citation](#), [abstract](#), [references](#), [index terms](#), [review](#)


We present a layered component model to support Web-based collaborative applications. We show how this model lets programmers focus on the particular logic of their applications, avoiding most of the issues related to collaboration, access control, and network management. The proposed model is organized into three layers on top of a foundation composed of commercial-off-the-shelf services and standard Internet protocols. The service level provides a network-transparent communications layer, data ...

10/797,258


**Keywords:** authoring tools, collaborative systems, educational web applications, learning technology standardization, web-based course delivery systems

4 A concurrency control theory for nested transactions (Preliminary Report)



 C. Beeri, P. A. Bernstein, N. Goodman, M. Y. Lai, D. E. Shasha  
August 1983 **Proceedings of the second annual ACM symposium on Principles of distributed computing PODC '83**

**Publisher:** ACM Press

Full text available:  [pdf\(1.72 MB\)](#) Additional Information: [full citation](#), [abstract](#), [references](#), [citations](#), [index terms](#)


Concurrency control is the activity of synchronizing transactions that access shared data. A concurrency control algorithm is regarded as correct if it ensures that any interleaved execution of transactions is equivalent to a serial one. Such executions are called serializable. Serializability theory provides a method for modelling and analyzing the correctness of concurrency control algorithms [BSW, Pa]. The concept of nested transaction has recently received mu ...

5 A database cache for high performance and fast restart in database systems



 Klaus Elhardt, Rudolf Bayer  
December 1984 **ACM Transactions on Database Systems (TODS)**, Volume 9 Issue 4

**Publisher:** ACM Press

Full text available:  [pdf\(1.72 MB\)](#) Additional Information: [full citation](#), [abstract](#), [references](#), [citations](#), [index terms](#), [review](#)

Performance in database systems is strongly influenced by buffer management and transaction recovery methods. This paper presents the principles of the database cache, which replaces the traditional buffer. In comparison to buffer management, cache management is more carefully coordinated with transaction management, and integrates transaction recovery. High throughput of small- and medium-sized transactions is achieved by fast commit processing and low database traffic. Very fas ...

6 A declarative approach to optimize bulk loading into databases



 Sihem Amer-Yahia, Sophie Cluet  
June 2004 **ACM Transactions on Database Systems (TODS)**, Volume 29 Issue 2

**Publisher:** ACM Press

Full text available:  [pdf\(1.00 MB\)](#) Additional Information: [full citation](#), [abstract](#), [references](#), [index terms](#)

Applications, such as warehouse maintenance, need to load large data volumes regularly. The efficiency of loading depends on the resources that are available at the source and at the target systems. Our work aims to understand the performance criteria that are involved in bulk loading data into a database and to devise tailored optimization strategies. Unlike commercial systems and previous research on the same topic, our approach follows the fundamental database principle of physical-logical ind ...


**Keywords:** Declarative bulk loading, algebra, recovery, side-effects

7 A deductive database audit trail



 David L. Sallach  
April 1992 **Proceedings of the 1992 ACM/SIGAPP Symposium on Applied computing: technological challenges of the 1990's**

**Publisher:** ACM Press

Full text available:  [pdf\(635.06 KB\)](#) Additional Information: [full citation](#), [references](#), [index terms](#)

8 A history of the Promis technology: an effective human interface

Jan Schultz

January 1986 **Proceedings of the ACM Conference on The history of personal workstations**

Publisher: ACM Press

Full text available: pdf(2.61 MB) Additional Information: [full citation](#), [abstract](#), [references](#), [index terms](#)

Scientific computing systems for individuals were pioneered early at Hewlett-Packard, beginning with the 9100A Desktop Calculator in 1968. Extensions of this first machine were soon seen in Personal Peripherals, such as Printers, Tape Cartridges, and Plotters, and followed by Graphic CRT Displays. By early 1972, the Desktop unit had been augmented by a very powerful Pocket Calculator, the ground-breaking HP 35A. This paper traces the evolution of these machines to the present day, ...

9 A quorum-consensus replication method for abstract data types

Maurice Herlihy

February 1986 **ACM Transactions on Computer Systems (TOCS)**, Volume 4 Issue 1

Publisher: ACM Press

Full text available: pdf(1.66 MB) Additional Information: [full citation](#), [abstract](#), [references](#), [citations](#), [index terms](#), [review](#)

Replication can enhance the availability of data in distributed systems. This paper introduces a new method for managing replicated data. Unlike many methods that support replication only for uninterpreted files, this method systematically exploits type-specific properties of objects such as sets, queues, or directories to provide more effective replication. Each operation requires the cooperation of a certain number of sites for its successful completion. A quorum for an operation is any s ...

10 A recovery algorithm for a high-performance memory-resident database system

Tobin J. Lehman, Michael J. Carey

December 1987 **ACM SIGMOD Record , Proceedings of the 1987 ACM SIGMOD international conference on Management of data SIGMOD '87**, Volume 16 Issue 3

Publisher: ACM Press

Full text available: pdf(1.50 MB) Additional Information: [full citation](#), [abstract](#), [references](#), [citations](#), [index terms](#)

With memory prices dropping and memory sizes increasing accordingly, a number of researchers are addressing the problem of designing high-performance database systems for managing memory-resident data. In this paper we address the recovery problem in the context of such a system. We argue that existing database recovery schemes fall short of meeting the requirements of such a system, and we present a new recovery mechanism which is designed to overcome their shortcomings. The proposed mecha ...

11 A reliable object-oriented data repository for a distributed computer system

Liba Svobodova

December 1981 **Proceedings of the eighth ACM symposium on Operating systems principles**

Publisher: ACM Press

Full text available: pdf(1.18 MB) Additional Information: [full citation](#), [abstract](#), [references](#), [citations](#), [index terms](#)

The repository described in this paper is a component of a distributed data storage system for a network of many autonomous machines that might run diverse applications. The repository is a server machine that provides very large, very reliable long-term storage for both private and shared data objects. The repository can handle both very small and very large data objects, and it supports atomic update of groups of objects that

might be distributed over several repositories. Each object is ...

**Keywords:** Atomic update, Crash recovery, Distributed data storage system, Memory management, Optical disk, Server, Stable storage


## 12 A simple and efficient implementation of a small database



A. Birrell, M. Jones, E. Wobber

November 1987 **ACM SIGOPS Operating Systems Review , Proceedings of the eleventh ACM Symposium on Operating systems principles SOSP '87**, Volume 21 Issue 5

**Publisher:** ACM Press

Full text available:  [pdf\(689.45 KB\)](#) Additional Information: [full citation](#), [abstract](#), [references](#), [citations](#), [index terms](#)

This paper describes a technique for implementing the sort of small databases that frequently occur in the design of operating systems and distributed systems. We take advantage of the existence of very large virtual memories, and quite large real memories, to make the technique feasible. We maintain the database as a strongly typed data structure in virtual memory, record updates incrementally on disk in a log and occasionally make a checkpoint of the entire database. We recover fr ...


## 13 A taxonomy of computer program security flaws



Carl E. Landwehr, Alan R. Bull, John P. McDermott, William S. Choi

September 1994 **ACM Computing Surveys (CSUR)**, Volume 26 Issue 3

**Publisher:** ACM Press

Full text available:  [pdf\(3.81 MB\)](#) Additional Information: [full citation](#), [abstract](#), [references](#), [citations](#), [index terms](#), [review](#)

An organized record of actual flaws can be useful to computer system designers, programmers, analysts, administrators, and users. This survey provides a taxonomy for computer program security flaws, with an Appendix that documents 50 actual security flaws. These flaws have all been described previously in the open literature, but in widely separated places. For those new to the field of computer security, they provide a good introduction to the characteristics of security flaws and how they ...

**Keywords:** error/defect classification, security flaw, taxonomy


## 14 A theory of reliability in database systems



Vassos Hadzilacos

January 1988 **Journal of the ACM (JACM)**, Volume 35 Issue 1

**Publisher:** ACM Press

Full text available:  [pdf\(1.88 MB\)](#) Additional Information: [full citation](#), [abstract](#), [references](#), [citations](#), [index terms](#), [review](#)

Reliable concurrent processing of transactions in a database system is examined. Since serializability, the conventional concurrency control correctness criterion, is not adequate in the presence of common failures, another theory of correctness is proposed, involving the concepts of commit serializability, recoverability, and resiliency.

## 15 Academic papers: Incorporating security components into database courses



Mario Guimaraes, Herb Mattord, Richard Austin

October 2004 **Proceedings of the 1st annual conference on Information security curriculum development**

**Publisher:** ACM Press

Full text available:  [pdf\(73.81 KB\)](#) Additional Information: [full citation](#), [abstract](#), [references](#), [index terms](#)

This paper describes information security topics to be presented in a course that provides instruction on the principles of database technology. Besides the customary coverage of securing the contents of databases (which is often presented in most courses of this type) we propose that is necessary to include instruction on other topics, such as securing the DBMS software, patch and version management of the DBMS application itself, issues and best practices surrounding security of database enable ...

**Keywords:** DBA, DBMS, ODBC, SQL, authentication, authorization

## 16 Adaptive Mechanisms and Policies for Managing Cache Hierarchies in Chip Multiprocessors



Evan Speight, Hazim Shafi, Lixin Zhang, Ram Rajamony

May 2005 **ACM SIGARCH Computer Architecture News , Proceedings of the 32nd Annual International Symposium on Computer Architecture ISCA '05**, Volume 33 Issue 2

**Publisher:** IEEE Computer Society, ACM Press

Full text available: [pdf\(128.39 KB\)](#) Additional Information: [full citation](#), [abstract](#), [citations](#), [index terms](#)

With the ability to place large numbers of transistors on a single silicon chip, manufacturers have begun developing chip multiprocessors (CMPs) containing multiple processor cores, varying amounts of level 1 and level 2 caching, and on-chip directory structures for level 3 caches and memory. The level 3 cache may be used as a victim cache for both modified and clean lines evicted from on-chip level 2 caches. Efficient area and performance management of this cache hierarchy is paramount given th ...

## 17 Adaptive, fine-grained sharing in a client-server OODBMS: a callback-based approach



Markos Zaharioudakis, Michael J. Carey, Michael J. Franklin

December 1997 **ACM Transactions on Database Systems (TODS)**, Volume 22 Issue 4

**Publisher:** ACM Press

Full text available: [pdf\(441.80 KB\)](#) Additional Information: [full citation](#), [abstract](#), [references](#), [citations](#), [index terms](#), [review](#)

For reasons of simplicity and communication efficiency, a number of existing object-oriented database management systems are based on page server architectures; data pages are their minimum unit of transfer and client caching. Despite their efficiency, page servers are often criticized as being too restrictive when it comes to concurrency, as existing systems use pages as the minimum locking unit as well. In this paper we show how to support object-level locking in a page-server context. Sev ...

**Keywords:** cache coherency, cache consistency, client-server databased, fine-grained sharing, object-oriented databases, performance analysis

## 18 An analysis of database workload performance on simultaneous multithreaded processors



Jack L. Lo, Luiz André Barroso, Susan J. Eggers, Kourosh Gharachorloo, Henry M. Levy, Sujay S. Parekh

April 1998 **ACM SIGARCH Computer Architecture News , Proceedings of the 25th annual international symposium on Computer architecture ISCA '98**, Volume 26 Issue 3

**Publisher:** IEEE Computer Society, ACM Press

Full text available: [pdf\(1.57 MB\)](#) [Publisher Site](#) Additional Information: [full citation](#), [abstract](#), [references](#), [citations](#), [index terms](#)

Simultaneous multithreading (SMT) is an architectural technique in which the processor


issues multiple instructions from multiple threads each cycle. While SMT has been shown to be effective on scientific workloads, its performance on database systems is still an open question. In particular, database systems have poor cache performance, and the addition of multithreading has the potential to exacerbate cache conflicts. This paper examines database performance on SMT processors using traces of th ...

#### 19 An early report on encompass

R. B. Terwilliger, R. H. Campbell

April 1988 **Proceedings of the 10th international conference on Software engineering**

**Publisher:** IEEE Computer Society Press

Full text available:  [pdf\(1.45 MB\)](#) Additional Information: [full citation](#), [abstract](#), [references](#), [citations](#), [index terms](#)

ENCOMPASS is an environment to support the incremental construction of Ada® programs using executable specifications and formal techniques similar to the Vienna Development Method. ENCOMPASS supports the rigorous development of software: parts of a project may use completely formal methods, while other, less critical parts use less expensive techniques. ENCOMPASS provides automated support for all aspects of the development process including specification, prototyping, ...


#### 20 An efficient and flexible method for archiving a data base



C. Mohan, Inderpal Narang

June 1993 **ACM SIGMOD Record , Proceedings of the 1993 ACM SIGMOD international conference on Management of data SIGMOD '93**, Volume 22 Issue 2

**Publisher:** ACM Press

Full text available:  [pdf\(969.13 KB\)](#) Additional Information: [full citation](#), [abstract](#), [references](#), [citations](#), [index terms](#)

We describe an efficient method for supporting incremental and full archiving of data bases (e.g., individual files). Customers archive their data bases quite frequently to minimize the duration of data outage. Because of the growing sizes of data bases and the ever increasing need for high availability of data, the efficiency of the archive copy utility is very important. The method presented here minimizes interferences with concurrent transactions by not acquiring any locks on the data b ...

Results 1 - 20 of 200

Result page: [1](#) [2](#) [3](#) [4](#) [5](#) [6](#) [7](#) [8](#) [9](#) [10](#) [next](#)

The ACM Portal is published by the Association for Computing Machinery. Copyright © 2006 ACM, Inc.

[Terms of Usage](#) [Privacy Policy](#) [Code of Ethics](#) [Contact Us](#)

Useful downloads:  [Adobe Acrobat](#)  [QuickTime](#)  [Windows Media Player](#)  [Real Player](#)